CHAOS
Theory and Applications
in Applied Sciences and Engineering

e-ISSN: 2687-4539
RESEARCH ARTICLE
Vol.5 / No.3 / 2023 / pp.219-232
https://doi.org/10.51537/chaos.1285094

Analyses of Reconfigurable Chaotic Systems and their Cryptographic S-box Design Applications

Mangal Deep Gupta [*,1], **Rajeev Kumar Chauhan** [α,2] **and Vipin Kumar Upaddhyay** [β,3]

[*]Department of Electronics and Communication Engineering, University Institute of Engineering & Technology, Babasaheb Bhimrao Ambedkar Central University, Lucknow, Uttar Pradesh, India, [α]Department of Electronics and Communication Engineering, MMMUT, Gorakhpur, Uttar Pradesh, India, [β]Electronics Engineering, Harcourt Butler Technical University, Kanpur, Uttar Pradesh, India.

ABSTRACT This manuscript includes the design and evaluation of the new four 16×16 S-boxes for subbyte operation in image encryption applications and estimates their strength using the following parameters: Dynamic Distance, BIC non-linearity, Bijective, Non-linearity, Strict Avalanche Criterion (SAC), and Balanced criterion. The S-box matrix is designed by a new reconfigurable 3D-Chaotic PRNG. This PRNG is designed using four different 3D chaotic systems i.e. Lorenz, Chen, Lu, and Pehlivan's chaotic systems. This reconfigurable architecture of PRNG exploits the ODEs of these four attractors that fit all four chaotic systems in a single circuit. The first part of this manuscript is focused to develop hardware-efficient VLSI architecture. To demonstrate the hardware performance, the PRNG circuit is implemented in Virtex-5 (XC5VLX50T) FPGA. A performance comparison of proposed and existing PRNGs (in terms of timing performance, area constraint, power dissipation and statistical testing) has been presented in this work. The PRNG generates the 24-bit random number at 96.438-MHz. The area of FPGA is occupied by only 16.66 %, 1.08%, 0.33 %, and 1.15% of the available DSP blocks, slice LUTs, slice registers and slices respectively. The designed S-boxes using reconfigurable PRNG fulfill the following criteria: Dynamic Distance, BIC non-linearity, Bijective, Non-linearity, Strict Avalanche Criterion (SAC), and Balanced criterion.

KEYWORDS
Cryptography
Chaotic systems
PRNG
Operating frequency
NIST Tests
S-Boxes
FPGA

INTRODUCTION

Random number generators are one of the essential components in cryptography, testing of VLSI circuits, bank transactions, financial market, avionics communications, etc. Random keys are required in various steps of cryptography like subbyte operation using S-box, encryption, decryption, etc. (Lambić and Nikolic 2019; ElSafty *et al.* 2021; Garcia-Bosque *et al.* 2018; Garipcan and Erdem 2020). Nowadays, smart systems that are used in the automation of houses and buildings, industry, energy, medical, transportation, communication system, etc. require the security of data transfer and Internet of Things (IoT) applications (G. Di Patrizio Stanchieri and Faccio 2019). Multimedia data such as video, image, audio and text can be communicated over the network very hugely but these shared data have a serious security concern. The general way to achieve this request is to design complex software or/and hardware-based systems, which can generate random sequences that provide the private and public keys to get the effective data encryption and decryption process.

In general, there are two types of PRNG: (1) Linear and (2) Nonlinear PRNG. Nonlinear PRNG is designed using nonlinear dynamical systems that exhibit chaos behaviour (L'Ecuyer 2012). In these types of systems, extreme sensitivity with the initial conditions causes chaotic behaviors over long-term randomness or unpredictability (H. S. Alhadawi and Lambi 2019). So, the chaotic system determines the nonlinear system with high randomness characteristics and low design cost. This makes it suitable for the designing of nonlinear PRNG. For designing a chaos-based cipher, a plain message is masked or encrypted using random keys (which is generated from chaotic maps) (Ü. Çavuşoğlu and Kaçar 2019; Wang *et al.* 2016). Chaotic systems generate a pseudorandom sequence, which can be applied in designing cryptographic

Manuscript received: 18 April 2023,
Revised: 6 August 2023,
Accepted: 28 October 2023.

[1]mangaldeepgcet@gmail.com (**Corresponding author**).
[2]rkchauhan27@gmail.com
[3]vipin08120@gmail.com

keys to get their valuable characteristics like random behavior, sensitivity to the initial conditions, and ergodicity (Li *et al.* 2001). So, the cryptographic properties of chaotic-map-based random sequences are very crucial from a security point of view for encryption algorithms. The idea of utilizing a 3D chaotic attractor for the designing of the PRNG is based on its ability that can generate a sequence of random numbers (X. Y. Wang and Kadir 2010; Artuğer and Özkaynak 2022b).

For the last 40 years, various simple chaotic systems have been found and continue the studied within the 3D quadratic autonomous framework. There are four criteria for the existence of chaotic behavior in the study of dynamic nonlinear systems (Pehlivan and Uyaroğlu 2012). The first well-known criterion is Lyapunov exponents (Wolf *et al.* 1985). It decides the chaotic behavior of dynamic systems. If at least one positive Lyapunov exponent presents in the dynamic system, the dynamic of this system is chaotic. The second criterion is Melnikov's. It is used to investigate the occurrence of chaotic behavior in Hamiltonian systems and it analyzes by estimating the distance between unstable and stable manifolds (Xu *et al.* 2009). The third one is Sil'nikov's criterion (T. Zhou and Čelikovský 2005). The last criterion is the topological horseshoes theory; it is based on some subsets of interest in the state space of continuous maps (Li and Yang 2010). These four criteria have been fulfilled by Lorenz (Lorenz 1963), Chen & Gupta (Gupta and Chauhan 2022, 2020), Lu (Lu and Chen 2002), and Pehlivan (Pehlivan and Uyaroğlu 2010) chaotic attractors.

The first 3D chaotic system was founded by Lorenz in 1963, it is a third-order autonomous system that displays very complex dynamic behaviors (Lorenz 1963). Another similar chaotic attractor was found by Chen in 1999. It is dual to the Lorenz system and topologically non-equivalent 3D chaotic system that shows interesting characteristics (Gupta and Chauhan 2022). Lu *and Chen* found another chaotic attractor known as Lu 3D chaotic system (Lu and Chen 2002). It represents the transition between Chen and Lorenz 3D attractors. It is important to note that the 3D chaotic attractors i.e. Lorenz (Lorenz 1963; Artuğer and Özkaynak 2022a), Chen (Gupta and Chauhan 2022), and Lu chaotic system (Lu and Chen 2002), have three particular fixed points: one saddle-foci and two unstable saddle-foci. Recently, Pehlivan *et al*. introduced a new 3D chaotic attractor (Pehlivan and Uyaroğlu 2010). It is similar to the Lorenz and Chen systems, but it includes six terms with two quadratics in a form and they have two very different fixed points (*i.e.* two stable node-foci).

The Lorenz, Chen, Lu, and Pehlivan chaotic attractors have been utilized in cryptography as PRNGs (Akgul *et al.* 2019; Alçın *et al.* 2016) due to their advantageous properties as discussed. To model the mathematical formation of a chaotic system, an ordinary differential equation (ODE) is used. It represents the rate-of-change of variables of a chaotic system. The ODEs can be solved using three different techniques i.e. Runge-Kutta, mid-point, or Euler's method (Zidan *et al.* 2011). Each chaotic system has a certain parameter value, which leads to the desired behavior of a chaotic system. One method to see the chaotic behavior of dynamic systems is to draw a three-dimensional (3D) plot, which is also known as an attractor. It demonstrates how the solutions of system variables evolve. Various analog and digital encryption circuits/systems have been designed using different chaotic attractors (Alawida *et al.* 2020; Zamli *et al.* 2023; Zhao *et al.* 2019; Rezk *et al.* 2020; Garcia-Bosque *et al.* 2019).

The subbyte operation in image encryption algorithms is the first step and primarily it decides the security strength of encrypted images. This operation is performed by the S-Box matrix (Zahid *et al.* 2021; Ahmad and Alsolami 2020; Alhadawi *et al.* 2020). It includes the 8-bit integers in random order in the form of a matrix. Therefore, the S-box plays the important role in image encryption algorithms. There is various image encryption algorithms available in the literature which shows the importance of S-boxes. The image encryption method using a chaotic attractors-based S-box matrix was proposed by Tang et. al. in (Tang *et al.* 2005). The S-box-based encryption using tent maps chaotic system was proposed by Y. Wong et. al. in (Wang *et al.* 2009). *M. Khan et. al.* proposed the new S-boxes using a Boolean function of a chaotic system (Khan *et al.* 2016, 2022). *Unal Çavusoglu et. al.* developed the chaotic S-box-based new image encryption algorithm which offers high-security strength and fast operation (Çavusoglu *et al.* 2017). The image encryption algorithm that uses different S-boxes in each cycle was proposed by Xiong Wang et. al. in (Wang *et al.* 2019; Artuğer 2023). The selection of S-boxes in this method is random which performs the image encryption.

This manuscript has introduced the four new S-boxes using reconfigurable PRNG. This reconfigurable PRNG is designed using four different 3D chaotic systems i.e. Lorenz, Chen, Lu, and Pehlivan attractors. All four chaotic systems reconfigure in a single architecture due to exploiting the similarities between the differential equations. The VLSI architecture of the proposed reconfigurable PRNG replaces the complex multiplication by hardwired shifting operation. The first part of this manuscript aims to develop hardware-efficient VLSI architecture that enhances the timing performances (in terms of latency, bit rate, and maximum operating frequency), length of the sequence, and randomness. The random sequences from all four chaotic systems are tested for randomness using the NIST test suite.

To evaluate the hardware performance, the proposed architecture has been implemented on prototype Virtex-5 (XC5VLX50T) FPGA. The next part of this manuscript includes the design of four new 16×16 S-boxes using the proposed reconfigurable PRNG. To check the suitability of proposed S-boxes in encryption applications, the following parameters: Dynamic Distance, Bijective, Balanced, Non-linearity, BIC non-linearity criterion and SAC have been evaluated in this manuscript. The remaining sections of this manuscript are arranged as follows: The dynamic behavior of Lorenz, Chen, Lu, and Pehlivan's chaotic systems are presented in Section-2. Section-3 includes the reconfigurable architecture of PRNG. The statistical description of generated bit Sequences using NIST is discussed in Section-4. A comprehensive description and comparison of PRNGs is presented in Section-5. Section-6 includes the design and evaluation of proposed S-boxes. The final conclusion of this manuscript is mentioned in Section-7.

DESCRIPTION OF LORENZ, CHEN, LU AND PEHLIVAN CHAOTIC SYSTEM

In this section, we construct parameter variables of Lorenz, Chen, Lu, and Pehlivan's three-dimensional (3D) chaotic attractors to design the hardware efficient and secure digital system of reconfigurable PRNG. The mathematical formation of chaotic attractors is done by ODEs. The numerical solution of ODEs can be done by three different methods: Runge-Kutta, Euler's method or mid-point. Hardware point of view, the most suitable approach is Euler's method. In this work, this method is adopted to solve the ODEs of a chaotic system. Eqs. (1) to (3) represent the Euler's equations corresponding variables: x_i, y_i and z_i.

$$x_{i+1} = x_i + h.\dot{x}_i \quad (1)$$

$$y_{i+1} = y_i + h.\dot{y}_i \quad (2)$$

$$z_{i+1} = z_i + h.\dot{z}_i \quad (3)$$

Table 1 to Table 4 includes the parameter values, range of variables and ODEs corresponding to Lorenz (Lorenz 1963), Chen (Gupta and Chauhan 2022), Lu (Lu and Chen 2002), and Pehlivan (Pehlivan and Uyaroğlu 2010) chaotic attractors. The selection of parameter values (as shown in Tables 1 to 4) offers hardware efficient reconfigurable architecture of PRNG. Table 1 shows the ODEs, range of variables, and parameter value for the Lorenz chaotic system.

Three variables of this chaotic system are represented by x_i, y_i and z_i, while a, b and c are the parameters. Similarly, Table 2 presents the ODEs, range of variables, parameter's value for Chen's chaotic system, where x_i, y_i and z_i, a, b and c show the same meaning. The third attractor is the Lu chaotic system. It has a wide range of parameter values in which the attractor displaces a different shape and represents the transition between Chen and Lorenz 3D attractors. The ODEs and range of variables are mentioned in Table 3, where a, b, c are the parameter variables. The last one is Pehlivan's chaotic system. It is similar to the Chen, and Lorenz systems, but it includes six terms with two quadratics in a form and they have two very different fixed points (i.e. two stable node-foci). Its ODEs are mentioned in Table 4, where a is the parameter variable, and x_i, y_i and z_i are system variables.

This section includes the simulation of the dynamic behavior of Lorenz, Chen, Lu, and Pehlivan's chaotic system using the Matlab Tool. To replace a large number of binary multiplication, parameter variables of chaotic systems are set to be specific values (as shown in Tables 1 to 4). The benefit of this approach is able to design multiplierless (except $x_i.y_i$ and $x_i.z_i$) reconfigurable digital chaotic PRNG. The plane and space plot of the proposed Lorenz, Chen, Lu, and Pehlivan's chaotic system are shown in Fig. 1. The Lorenz system has a 3D attractor as shown in Fig. 1(a), with parameters values: $a = 32, b = 4, c = 32$, initial condition $(x_0, y_0, z_0) = (1, 1, 1)$ and step size: $h = 2^{(-8)}$. Next, the 3D attractor of the Chen chaotic system is present in Fig. 1(b), with the parameters values: $a = 32, b = 4, c = 24$, initial condition $(x_0, y_0, z_0) = (5, -15, 40)$ and step size:$h = 2^{(-8)}$ Fig. 1(c) shows the chaotic attractor of Lu system with $a = 32, b = 4, c = 16$, initial condition $(x_0, y_0, z_0) = (1, 1, 1)$ and step size: $h = 2^{(-8)}$. Similarly, Fig. 1(d) represents the chaotic attractor of Pehlivan system with $a = 0.5, h = 2^{(}-8)$ and initial condition $(x_0, y_0, z_0) = (0.001, 0.001, 0)$. The phase plane behavior of Lorenz, Chen, Lu, and Pehlivan's chaotic system are shown in Fig. 2 to Fig. 5, correspondingly.

The xy, xz, and yz phase portraits of the Lorenz system are shown in Fig. 2 with the same parameter values and initial condition. The two-dimensional (2D) attractor plots in the plane of Chen's chaotic system are displayed (with the following details: parameter values $a = 32, b = 4, c = 24, h = 2^{-8}$ and initial condition: $(x_0, y_0, z_0) = (5, -15, 40)$ in Fig. 3. Similarly, Fig. 4 represents the phase portraits of Lu system with $a = 32, b = 4, c = 16, h = 2^{-8}$ and initial condition $(x_0, y_0, z_0) = (1, 1, 1)$. Finally, the xy,xz and yz phase portraits of the Pehlivan system with the same parameter value and initial condition (as discussed in Table 4) are shown in Fig. 5.

Table 1 Variables range and Parameter's value for Lorenz chaotic system.

Lorenz chaotic system		
ODEs Lorenz (1963)	Parameters	Range
$\dot{x}_i = a(y_i - x_i)$	$a = 32, b = 4, c = 32,$	$-28.1805 \leq x \leq 29.2467$
$\dot{y}_i = -x_i z_i + c x_i - y_i$	$h = 2^{-8}, x_0 = 1,$	$-31.1805 \leq y \leq 33.1210$
$\dot{z}_i = x_i y_i - b z_i$	$y_0 = 1, z_0 = 1$	$0.9215 \leq z \leq 58.6626$

Table 2 Variables range and Parameter's value for Chen's chaotic system.

Chen Chaotic System		
ODEs Gupta and Chauhan (2022)	Parameters	Range
$\dot{x}_i$=a.$(y_i - x_i)$	$a = 32, b = 4, c = 14,$	-24.280$\leq$ x $\leq$ 23.9385
$\dot{y}_i$=-$x_i.z_i$+(c-a).x_i+c.y_i	h=2^{-8}, $x_0 = 5,$	-27.4307$\leq$ y $\leq$27.0290
$\dot{z}_i$= $x_i.y_i$- b.z_i	$y_0 = -15, z_0 = 40$	1.7161$\leq$ z $\leq$ 47.230

Table 3 Variables range and Parameter's value for Lü chaotic system.

Lu Chaotic System		
ODEs Lu and Chen (2002)	Parameters	Range
$\dot{x}_i$=a.$(y_i - x_i)$	$a = 32, b = 4, c = 16,$	-20.8399$\leq$ x $\leq$ 21.2057
$\dot{y}_i$=-$x_i.z_i$+c.y_i	h=2^{-8}, $x_0 = 1,$	-22.8983$\leq$ y $\leq$23.3546
$\dot{z}_i$= $x_i.y_i$- b.z_i	$y_0 = 1, z_0 = 1$	0.8931$\leq$ z $\leq$ 34.5366

Table 4 Variables range and Parameter's value for Pehlivan's chaotic system.

Pehlivan Chaotic System		
ODEs Pehlivan and Uyaroğlu (2010)	Parameters	Range
$\dot{x}_i$=$y_i - x_i$	$a = 0.5, h = 2^{-8},$	-2.8411$\leq$ x $\leq$ 2.7743
$\dot{y}_i$=-$x_i.z_i$+a.y_i	$x_0 = 0.001, y_0 = 0.001,$	-4.7402$\leq$ y $\leq$4.8913
$\dot{z}_i$= $x_i.y_i$- a	$z_0 = 0$	-2.9902$\leq$ z $\leq$ 6.6909

PROPOSED DIGITAL ARCHITECTURE OF RECONFIGURABLE CHAOTIC PRNG

This section includes the VLSI circuit of reconfigurable chaotic PRNG using Lorenz, Chen, Lu, and Pehlivan 3D attractors. The general architecture has been constructed by the exploitation of similarity between all chaotic attractors which leads to fit into a single structure. The parameters of Lorenz system has been set to $(2^5, 2^2, 2^5, 2^{-8})$ corresponding (a, b, c, h). Moreover, Table 1 depicts the range of variables: $-28.1805 \leq x \leq 29.2467, -31.1805 \leq y \leq 33.1210$ and $0.9215 \leq z \leq 58.6626$. Similarly, Table 2 to Table 4 include the step size, parameters, and variable range of the system of Chen, Lu, and Pehlivan correspondingly. The benefits of this approach, all binary multiplication operations of ODEs and Euler's expressions (except $x_i.y_i$ and $x_i.z_i$) has been carried out by the operation of hardwire shifting rather than binary multiplication. In this modelling, 2's complement and the fixed-point scheme have been used in which 7 MSB represent the amount of integer including sign bit. On the other side, the rest 25 bits represent the fractional value of all parameters and variables. To retain the same fractional bits of 25, the truncation rounding scheme is performed in this operation.

This reconfigurable feature of PRNG is designed by hardwired shifting operations, additions, subtractions, and multiplexing schemes. Fig. 6 represents the VLSI architecture of proposed reconfigurable PRNG using Lorenz, Chen, Lu, and Pehlivan 3D attractors. This architecture offers the opportunity to configure the four different systems and it is controlled by a 2-bit signal which is denoted by *Confg*[1:0]. Pehlivan's chaotic system is configured by *Confg*[1:0]=2'b00, similarly, Lu chaotic system is configured by *Confg*[1:0]=2'b01. Similarly, when *Confg*[1:0] value is 2'b10, the multiplexer switches to the Lorenz system, while the value is 2'b11, architecture computes the Chen system for generating pseudorandom numbers. Three separate 32-bit register block of this figure is designed to evaluate the value of Euler's equations (as given in Eq. (1) to Eq. (3)). The initialization of registers corresponding to three variable is done by Reset signal which controls the 2×1-multiplexer, initially all registers hold the value of X_0, Y_0 and Z_0 correspondingly. The adder used in this block to add the present value of variables (X_i, Y_i, Z_i) with differential value $(h.X, h.Y, h.Z)$ as shown in blocks.

The computational process to evaluate differential value h.X is depicted in Block-1. It is required subtraction to subtract the value of X_i from Y_i. In this block, the logical OR value of *Confg*[1] and *Confg*[0] signal, act as a select line of 2×1-multiplexer. When the value of logic OR operation is '0', the multiplexer gives the differential value (h.X) of Pehlivan's chaotic system, which is the 8-bits hardwired left-shifted of subtracted value. While the value of logic OR operation is '0', the multiplexer gives the 3-bit left shifting of subtracted value as a differential value (h.X) corresponding to Lorenz, Chen, and Lu chaotic system.

The evaluation of h.Y according to the ODE of variable Y (corresponding Lorenz, Chen, Lu, and Pehlival chaotic systems) given in Block-2. In this block, 2-bit *Confg*[1:0] signal, act as a control signal of a 4×1-multiplexer. When the value of *Confg* signal is $2^{'b00}$, multiplexer passes the 9-bit hardwired left shifted value of Y_i according to Pehlivan's chaotic system. The multiplexer passes the 4-bit hardwired left shifted value of Y_i according to Lu, when the value of *Confg* signal is $2^{'b01}$. When the value of *Confg* signal is $2^{'b10}$, multiplexer passes the subtracted value (8-bit hardwired left shifted value of X_i from the 3-bit hardwired left shifted value of Y_i). When the value of *Confg* signal is $2^{'b11}$, multiplexer passes the computational value of $2^{-8}.(8.x_i + 24.y_i))$ according to Chen's chaotic system. One 32-bit binary multiplier is required in this block to multiply the value of Z_i with X_i. To subtract the multiplexer's output with an 8-bit left-shifted multiplier's output, one 32-bit subtractor is used as shown in the figure and their output gives the differential value (h.Y). Here, the shifting operation performs the multiplication operation which is not utilized any hardware resources.

Similarly, Block-3 presents the computational block to evaluate the differential value (h.Z). Here, the logical OR value of *Confg*[1] and *Confg*[0] act as a control signal of the multiplexer. It passes the

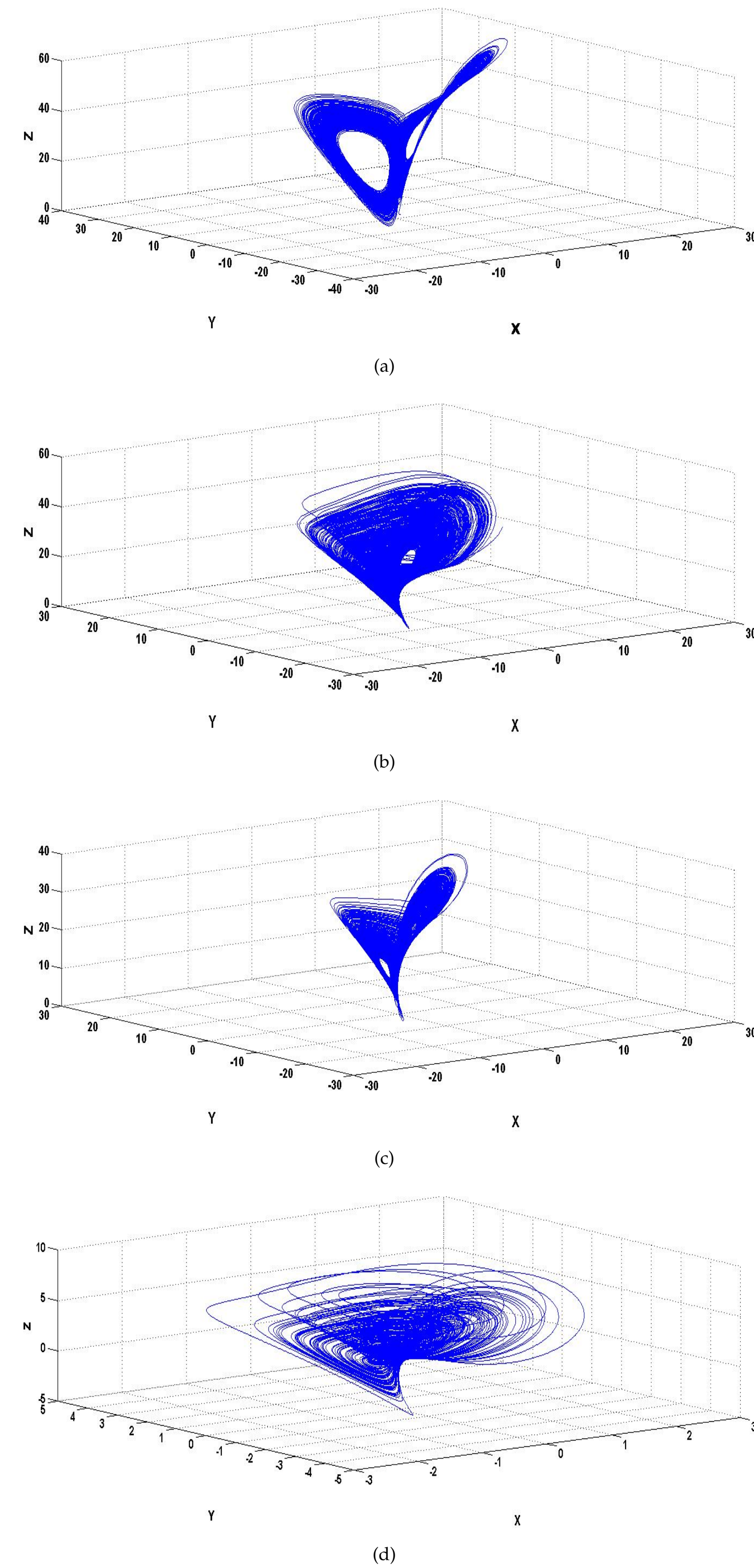

Figure 1 Chaotic attractor in the plane of: (a) Lorenz; (b) Chen; (c) Lu; and (d) Pehlivan systems.

value $2^{(-9)}$, when the control signal is equal to logic '0'. While, for control signal equal to logic "1", multiplexer pass the 6-bits left shifted value of Z_i. This block includes one 32-bit binary multiplier

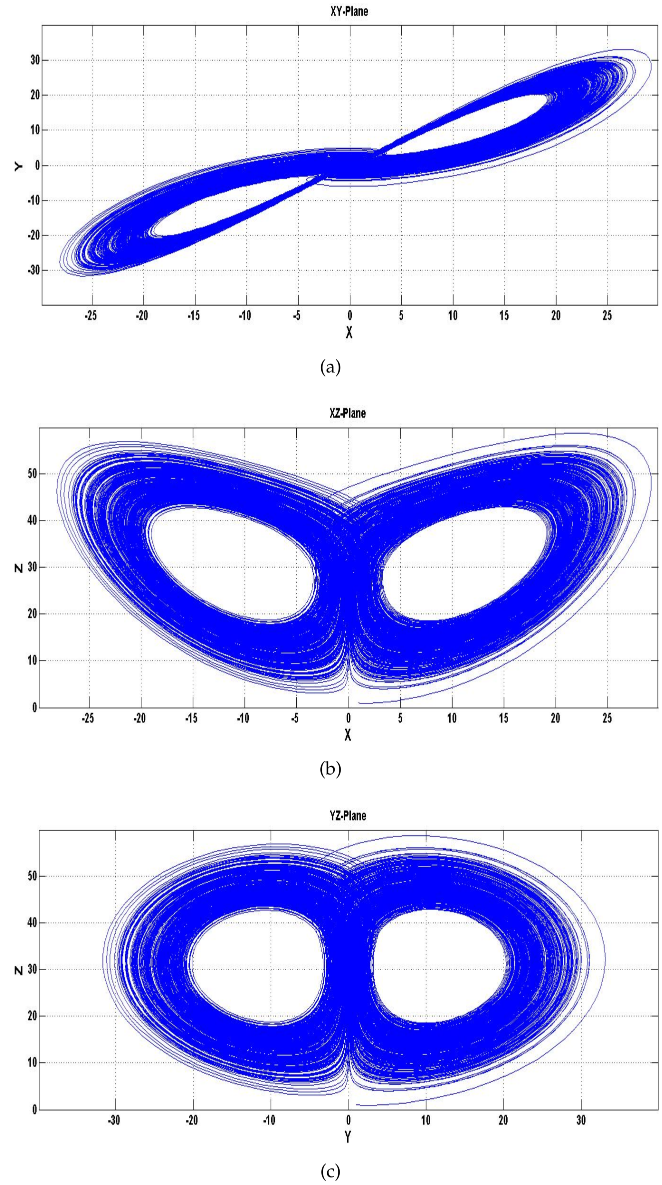

Figure 2 Chaotic attractor in plane of Lorenz system with , $h = 2^{-8}$, $a = 32$, $b = 4$, $c = 32$ and initial condition (x_0, y_0, z_0) = (1, 1, 1): (a) x-y plane; (b) x-z plane; (c) y-z plane.

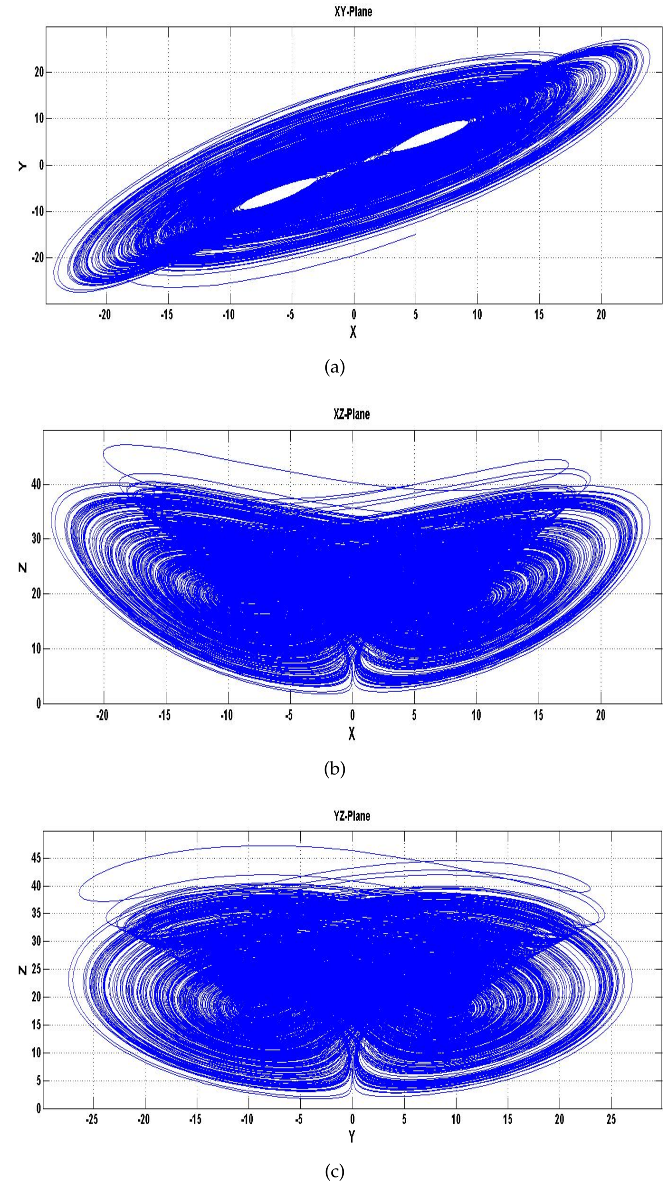

Figure 3 Chaotic attractor in plane of Chen's system with , $h = 2^{-8}$, $a = 32$, $b = 4$, $c = 24$ and initial condition (x_0, y_0, z_0) = (5, -15, 40): (a) x-y plane; (b) x-z plane; (c) y-z plane.

that multiplies the 32-bit value of Y_i with X_i. The subtraction circuit is also used in this block that subtracts the multiplexer's output with the 8-bit left-shifted of multiplier's output, which gives the differential value h.Z . The output of this block generates the 24-bit random numbers in each iteration. These 24-bit data is captured from 8 Least Significant Bits (LSBs) from each chaotic variable.

Example of the Proposed reconfigurable PRNG: Let $a = 32$, $b = 4$, $c = 24$, $h = 2^{(-8)}$, X_0=5 (00001010000000000000000000000000), Y_0=-15 (11100010000000000000000000000000), Z_0=40 (01010000000000000000000000000000) and *Confg*=3. When the *Confg* value is 2'b11, architecture computes the Chen system for generating pseudorandom numbers. Block-1 generates the differential value: $h.(X_0)$ =11111111011000000000000000000000, Block-2 generates the differential value: $h.(Y_0)$ =11111111110101111111110011100000, and Block-3 generates the differential value: $h.(Z_0)$=11111111111110101111111011010100.

The value of X_1=00001001011000000000000000000000, Y_1=11100001110101111111110011100000, and Z_1=01001111111110101111111011010100 have been generated from three Euler's blocks separately. Finally, captured the 8 Least Significant Bits (LSBs) of each chaotic variable: X_1=00000000, Y_1=11100000 and Z_1=11010100, this architecture generates a 24-bits pseudo-random number in 1^{st} iteration: OUT_1=000000001110000011010100. Similarly, OUT_2=000000001100000010101000, OUT_3= 111100111100111011011110 and so on, generate in the next iterations.

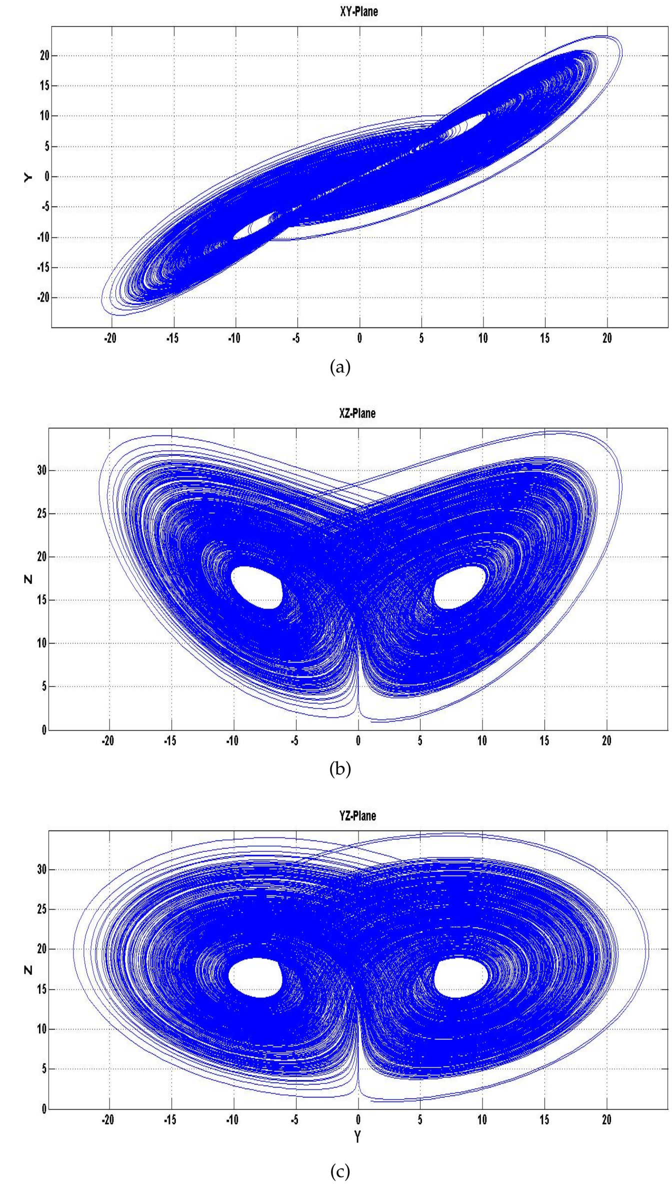

Figure 4 The chaotic attractor in the plane of Lu system with, $h = 2^{-8}$, $a = 32$, $b = 4$, $c = 16$ and initial condition $(x_0, y_0, z_0) = (1, 1, 1)$: (a) x-y plane; (b) x-z plane; (c) y-z plane.

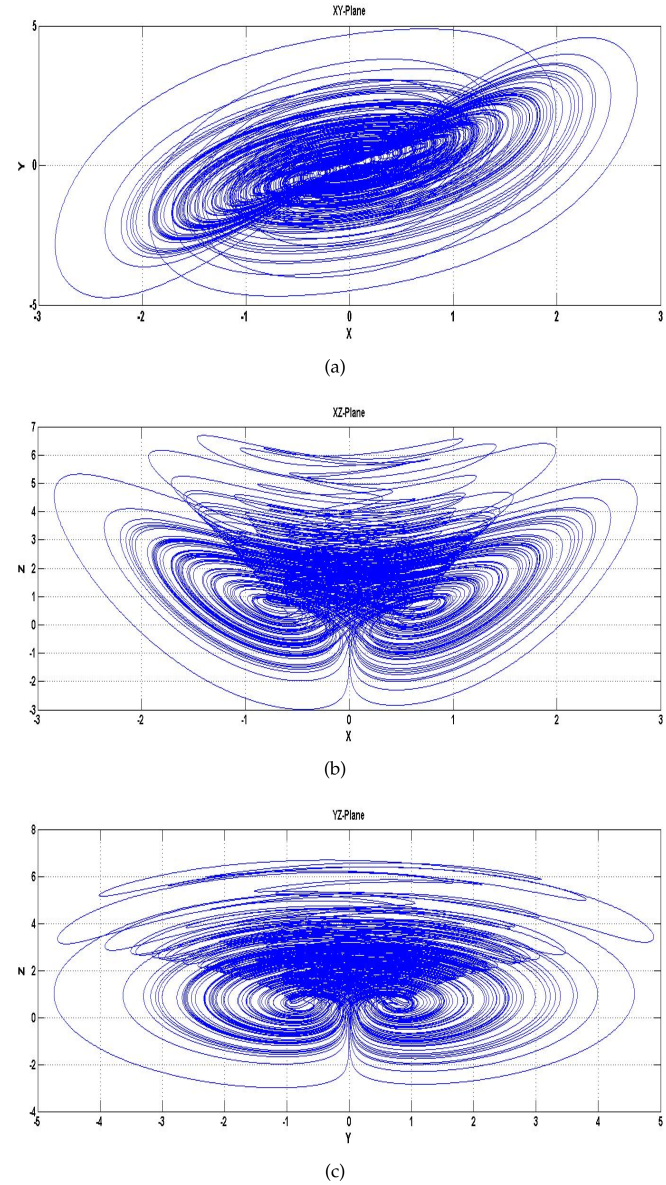

Figure 5 Chaotic attractor of Pehlivan system with $a = 0.5$, initial condition $(x_0, y_0, z_0) = (0.001, 0.001, 0)$ and $h = 2^{-8}$: (a) x-y plane; (b) x-z plane; (c) y-z plane.

IMPLEMENTATION OF 32-BIT PRNG AND STATISTICAL TESTS

The implementation of 32-bit PRNG circuits is done on Virtex-5 FPGA (XC5VLX110T). Its synthesis has been done on the ISE design suite by Xilinx. Initially, its Register Transfer Level (RTL) design is done using Verilog HDL. Table 6 depicts the hardware performance including the parameters: area constraint (in terms of slice look-up-tables (LUTs), occupied slices and slice registers), Digital signal processing (DSP) blocks, timing performance (in terms of critical path delay and maximum operating frequency), and power dissipation per unit frequency. The post-layout simulation waveform of proposed PRNGs are shown in Fig. 7(a), 7(b), 7(c), and 7(d) corresponding to four different configurations i.e. Pehlivan, Lu, Lorenz, and, Chen's PRNG.

The post routing simulation waveform of 32-bit Pehlivan's chaotic system-based PRNG is shown in Fig. 7(a). The control signal *(Confg)* is used to configure the systems, when its value is equal to 00, it configures Pehlivan's chaotic system. This simulation takes the initial value: $(X_0, Y_0, Z_0) = (0.96248769, 1.20541650, 42.13836362)$. The signal "CLK" and "Reset" are the master clock signal and reset signal respectively. Initialization of the registers with X_0,Y_0, and Z_0 is done by "Reset" signal. The three variable X_i[32:0], Y_i[32:0] and Z_i[32:0] represent the iterative values. Its 8-bit LSBs segments combine to generate a 24-bit pseudo-random number, which is given by the variable *OUT[23:0]*.

Similarly, Fig. 7(b), 7(c), and 7(d) show the post routing simulation waveform of 32-bits reconfigurable PRNG for Lu, Lorenz, and Chen 3D attractors with *Confg[1:0]* equal to 2'b01, 2'b10 and 2'b11 correspondingly. This simulation takes the initial value:

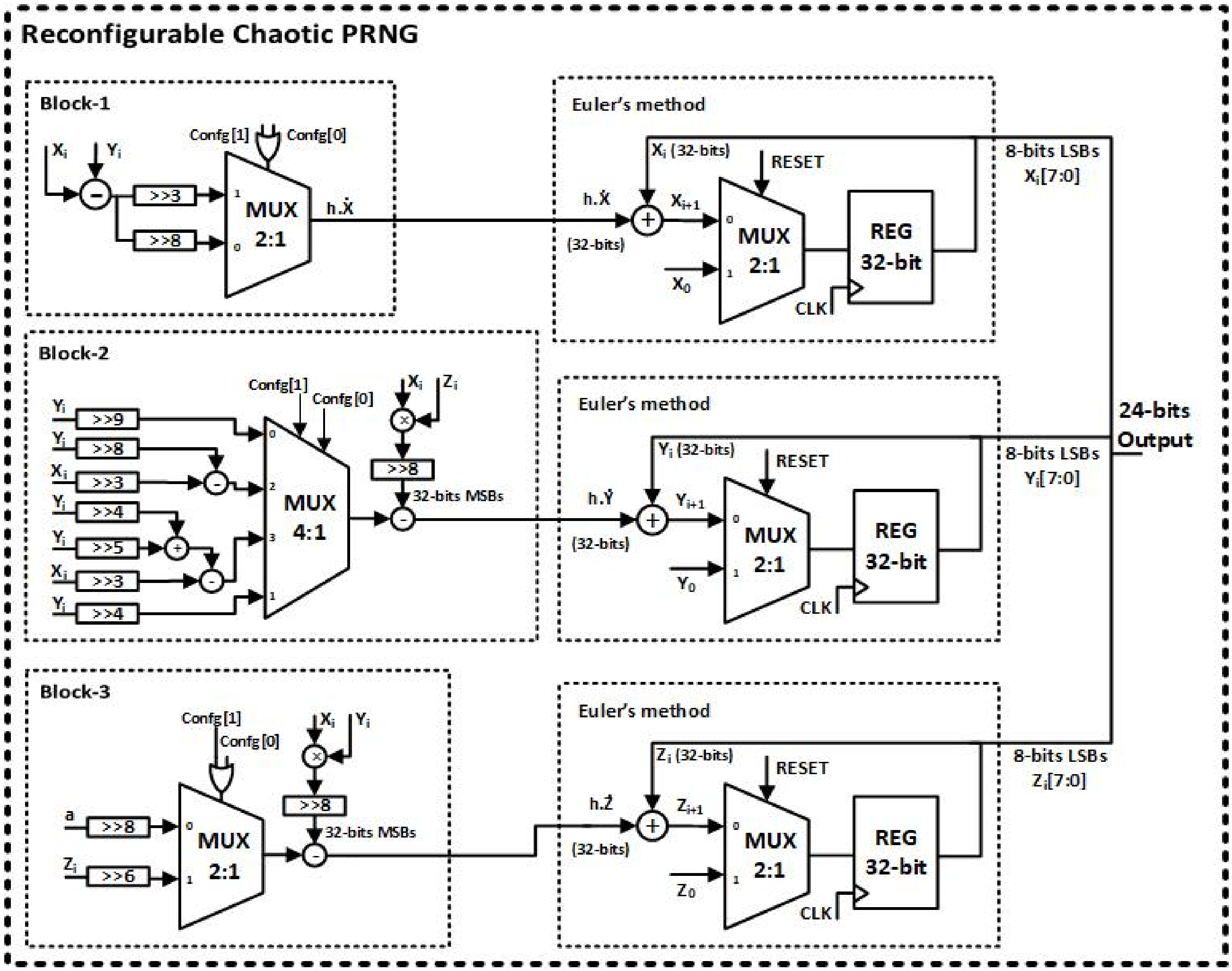

Figure 6 Proposed architecture of reconfigurable chaotic PRNG using Lorenz, Chen, Lu, and Pehlivan chaotic systems.

$(X_0, Y_0, Z_0) = (1,1,1), (1,1,1)$, and $(5, -15, 40)$ respectively. In this figure, the *"CLK"* and *"Reset"* signals represent the same meaning. Similarly, the three variable X_i[32:0], Y_i[32:0], and Z_i[32:0] represent the iterative values. Its 8-bit LSBs segments combine to generate 24-bits pseudo-random numbers, which are given by the variable *OUT[23:0]*.

The proposed reconfigurable PRNG demonstrates over the existing architectures of PRNGs. It provides the opportunity to switch between four different 3D-Chaotic systems. This architecture is a completely digital circuit, which is easily suitable for real-time digital applications where PRNG is required. The comparison table of the hardware performance and security strength is given in Table 6. This table summarizes the NIST results, timing performance, power consumption, and area resources.

The maximum operating frequency of proposed PRNG is increased by 23.40% as compared with PRNG (Rezk *et al.* 2019), while it increases by 3.69% as compared with PRNG based on logistics (Pande and Zambreno 2013). A resources of FPGA (in terms of occupied slices, slice registers, slice LUTs, and DSP blocks) is utilized by designed PRNG circuit is slight increases (as compared with existing literature) due to the involvement of four different chaotic systems in a single architecture. However, it is suitable for generating a high degree of randomness and large period pseudorandom numbers. The proposed architecture consumes 8.6125 mW/MHz total power on Virtex-5 for a 32-bit design. The statistical analysis of generated keys has been done by the NIST test suit. This result also depicts that the security strength of keys from four different configurations is highly secure and it can be used in S-box generation, image encryption, etc.

The statistical testing of a random number generator is federal information, which processes the standard issued by the NIST (Rukhin *et al.* 2000). This test includes the fifteen different statistical tests that perform to check the security strength of generated random sequences in all aspects of security. For this test, we take 100 samples of bit sequences (each sample has a 10^6 random bits sequence). The NIST benchmark test of these four sequences has been performed. This test suite set the level of significance equal to 0.01. This means that the resulting p-value of each sample should be greater than or equal to the level of significance for indicating the randomness strength of generated bit sequences. The sequences have been generated using parameters and initial seed values as mentioned in Table 1 to Table 4. The four different generated sequences from the proposed reconfigurable PRNG have been passed all the tests. Table 5 present the proportional value and maximum p-value corresponding to each test of NIST. This table depicts that test sequences pass all fifteen test of NIST, which indicate the high security strength of generated random sequences from the proposed PRNG circuit.

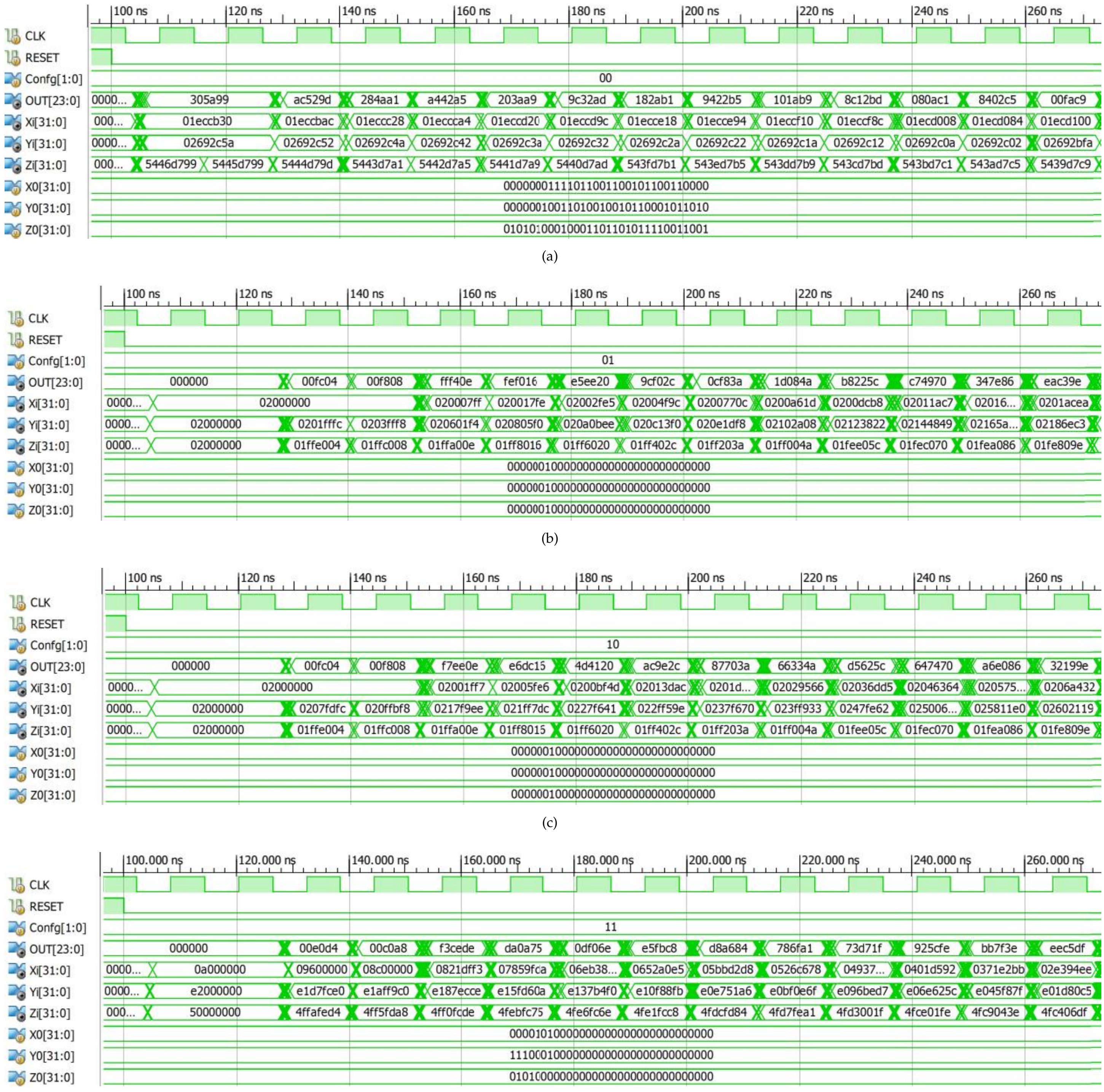

Figure 7 Post routing simulation waveform of proposed 32-bit reconfigurable chaotic PRNG: (a) Pehlivan; (b) LU; (c) Lorenz and (d) Chen system.

Table 5 FPGA synthesis result of proposed and existing architecture of Chaotic-based PRNG

	Proposed	(Zidan et al., 2011)	(de la Fraga et al., 2017)	(Rezk et al., 2019)	(Pande & Zambreno, 2013)
Chaotic System	**(Lorenz + Chen + Lu + Pehlivan)**	**Lorenz & Bernoulli**	**(Lu + Lorenz)**	**Logistic**	
Operand Size	32-bits	32-bits	32-bits	32-bits	32-bits
Number of 3D chaotic attractors	4	1	1	2	1
FPGA	Virtex 5 (XC5VLX50T)	Virtex 4 (XC4VSX35)	Spartan 3E (XC3S500E)	Virtex 5 (XC5VLX50T)	Virtex 6 (XC6VLX75T)
Occupied Slices/Total	83/7200	145/15360	342/7200	100/7200	181/11640
Slice registers/Total	96/28800	96 /30,720	108/28,800	96 /28800	160/93120
Slice LUTs/Total	313/28800	287 /30,720	575/28,800	276/28800	643/46560
DSP blocks/Total	8/48	8/192	9/48		16/288
Frequency (MHz)	96.438	53.53	36.90		93.00
NIST	Pass	–	–		–

Table 6 NIST Test Results

Test	Lorenz (10)		Chen (11)		Lu (01)		Pehlivan (00)	
	P-value within success sequence	Proportion successful out of 100	P-value within success sequence	Proportion successful out of 100	P-value within success sequence	Proportion successful out of 100	P-value within success sequence	Proportion successful out of 100
Frequency Test within a Block	0.961876	98	0.905225	99	0.998261	96	0.802587	99
Frequency (Monobit)	0.719747	99	0.657933	100	0.888660	96	0.841481	98
Runs Test	0.955825	99	0.474986	100	0.639464	98	0.996907	99
Longest-Run-of-Ones in a Block	0.844731	99	0.719747	97	0.951366	99	0.942871	96
Linear Complexity	0.657933	98	0.699313	98	0.798139	97	0.933026	98
Binary Matrix Rank	0.862457	99	0.949536	99	0.949536	98	0.862457	97
Approximate Entropy	0.534146	98	0.574903	99	0.153763	98	0.999952	100
Discrete Fourier Transform	0.657933	99	0.926884	96	0.771671	97	0.646355	99
Overlapping Template Matching	0.822183	100	0.883171	100	0.856837	100	0.924076	97
Non-overlapping Template Matching	0.971699	98	0.851383	97	0.779188	99	0.798139	97
Cumulative Sums	0.554420	100	0.867692	100	0.762693	96	0.990843	98
Universal Statistical Test	0.498264	98	0.697354	100	0.802673	96	0.864253	100
Serial Test	0.042808	100	0.304126	100	0.759756	99	0.989703	98
	0.474986	99	0.946308	99	0.262249	99	0.653842	99
Random Excursions	0.867523	98	0.643582	99	0.943559	96	0.983256	100
Random Excursions Variant	0.578556	96	0.732568	99	0.969182	99	0.827614	96

DESIGN AND EVALUATION OF S-BOXES

This section designs the four different new S-box matrixes using the proposed reconfigurable PRNG. The steps for designing S-boxes from PRNG are illustrated: The first step is to segment the 24-bit random numbers into three parts and each 8-bit binary value is converted into decimal form. This decimal value compares with the existing value of the matrix in Step two and it includes the element of the matrix if the value is not repeated. This process is repeated until the entire matrix element is filled. And finally generates the S-boxes, which contain the 256 different 8-bit elements in random order. Tables 6, 7, 8 and 9 present the S-box matrix corresponding to *Confg* equal to 2′b00, 2′b 01, 2′b 10, and 2′b 11.

Since the critical part of cryptography is S-boxes thus, important characteristics of a cryptographically strong S-box have been examined in this section. The evaluated characteristics exhibit features like Average non-linearity of all Boolean functions, non-linearity of Boolean functions, Balanced, Bijective, Non-linearity of S-Box, BIC non-linearity criterion, Strict Avalanche Criterion (SAC), and Dynamic Distance. Moreover, Outcomes have been compared with other techniques reported in the literature. The reference of the all-mathematical definitions of the above-mentioned parameters is (Cassal-Quiroga and Campos-Cantón 2020; Ishfaq 2018; Gupta and Chauhan 2021).

It is well known that the criterion of bijective property of S-boxes is equivalent to $2^{n-1} = 128$ where $n = 8$. Since it satisfies the bijective criterion for all proposed S-boxes thus it is considered as desired value for the bijective criterion. Simultaneously, the balanced, one-to-one and surjective properties are also satisfied for the proposed S-boxes.

The non-Linearity criterion is another parameter that holds the nonlinearity property between the vector of input and output of S-boxes. It holds a better explanation for the dissimilarity degree between Boolean and linear functions (Cassal-Quiroga & Campos-Cantón, 2020). The calculation of eight Boolean functions of non-linearity property has been performed for the S-boxes. The calculated value of eight non linearity function of non-linearity property for the S-box-1 are 104, 106, 104, 102, 100, 102, 108 and 104, and for the S-box-2 are 104, 104, 104, 106, 106, 102, 104 and 104. In same way the eight non linearity Boolean values for S-box-3 and S-Box-4 are (102, 104, 106, 104, 110, 106, 106, 102) and (102, 104, 106, 104, 110, 106, 106 and 102) respectively. It is well-identified that larger non-linear values ensure the highest ability to resist

Table 7 S-Box-1 using proposed chaotic PRNG with *Confg* equal to 00.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	89	112	123	134	4	146	179	152	169	224	44	192	13	215	58	65
2	232	121	88	21	15	111	66	165	59	157	156	210	180	87	30	119
3	240	53	164	137	76	209	34	99	254	187	122	43	84	217	55	251
4	6	18	52	109	41	98	8	64	144	190	193	216	36	239	238	194
5	28	96	29	74	195	158	100	181	5	204	168	167	227	214	73	250
6	235	22	186	94	2	166	211	32	199	110	49	113	160	171	97	207
7	253	145	45	39	57	86	155	81	133	71	105	243	129	159	153	12
8	106	31	200	206	161	241	175	79	19	126	197	173	202	188	42	90
9	138	218	125	10	162	154	234	26	27	212	141	170	70	3	0	247
10	182	117	147	196	140	78	108	16	148	255	69	77	118	17	213	9
11	93	131	68	231	11	25	75	101	233	47	103	249	128	127	142	178
12	177	102	51	229	205	23	230	120	24	237	191	50	85	1	136	33
13	80	150	221	67	132	37	62	248	245	223	225	95	198	48	244	219
14	201	130	116	220	246	222	72	115	151	61	54	40	236	35	242	14
15	252	228	92	46	83	60	163	82	139	63	203	189	107	104	114	174
16	38	20	185	143	208	135	7	176	183	124	172	184	149	91	226	56

Table 8 S-Box-2 using proposed chaotic PRNG with *Confg* equal to 01

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	229	238	32	156	240	44	12	248	58	29	8	74	184	34	92	199
2	211	201	103	52	76	235	151	202	252	56	33	99	140	216	204	196
3	41	39	217	23	90	145	210	97	75	87	62	7	161	244	220	153
4	223	116	236	254	162	251	59	233	6	31	182	86	30	158	85	122
5	113	123	207	147	70	187	175	27	28	141	212	25	142	143	146	243
6	178	71	128	114	173	81	253	55	169	197	73	127	10	93	215	181
7	171	2	5	18	189	249	230	206	84	195	200	37	82	4	109	150
8	225	36	14	72	17	69	110	131	239	208	194	247	125	163	13	26
9	186	226	219	106	38	214	57	213	117	152	191	133	64	50	0	9
10	137	126	168	107	45	172	179	190	205	118	192	79	95	120	155	83
11	177	22	136	167	231	174	180	157	119	121	42	88	105	100	124	224
12	68	63	222	134	98	166	20	53	96	246	149	242	66	43	154	237
13	159	48	89	255	160	1	67	40	232	21	241	15	144	3	250	170
14	148	193	94	60	218	78	61	102	185	221	111	129	130	11	108	203
15	228	135	164	47	234	176	46	112	188	139	198	183	65	51	80	209
16	104	245	77	54	24	132	35	138	115	49	101	227	165	91	19	16

powerful attacks.

The randomness of the S-box is measured by Strict Avalanche Criterion (SAC). If there is an input change then random behavior comes into the picture which is regarded as the avalanche effect in S-box. There is an alteration in each output bit with one-half of the probability if any change is made in the single bit of input. This phenomenon reflects the Strict Avalanche Criterion (SAC). It is well known that there is a 50% dependency of Boolean function on each input bit for a better explanation of this criterion. The generated SAC values of S-box-1, -2, -3 and -4 are tabulated in Table [16, 17,18] respectively. The corresponding minimum, maximum, and average SAC values of 0.3606, 0.5938, and 0.500016 for S-box-1 have been obtained. In the same way, the corresponding minimum, maximum, and average SAC values of 0.3906, 0.6406, and 0.504894 for S-box-2 have been evaluated and for S-box-3 the minimum, maximum and average values are 0.3906, 0.5781, and 0.503669. At last, the minimum, maximum, and average values for S-box-

Table 9 S-Box-3 using proposed chaotic PRNG with *Confg* equal to 10

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	247	238	14	230	220	22	77	65	32	172	158	44	135	112	58	102
2	81	177	12	119	19	99	210	92	179	221	233	107	69	30	9	17
3	20	199	222	229	54	235	73	126	13	248	209	129	98	138	190	36
4	48	181	228	226	16	156	18	237	197	78	187	110	123	27	203	43
5	127	184	80	55	219	87	70	183	120	174	46	71	171	60	23	131
6	96	200	25	45	62	168	109	133	84	94	31	164	143	33	21	213
7	47	7	49	215	163	37	117	147	83	29	79	41	169	212	40	191
8	53	8	93	34	68	195	104	3	236	188	4	194	241	245	125	162
9	5	89	185	225	88	227	218	128	42	250	202	207	189	66	132	63
10	118	51	75	141	160	111	243	137	204	86	155	205	206	232	176	82
11	139	255	186	167	6	246	165	136	39	103	114	211	214	244	192	208
12	28	239	253	0	61	242	100	251	57	101	157	161	152	148	52	216
13	145	249	170	154	113	142	178	124	90	105	151	15	224	56	182	72
14	64	134	140	97	91	35	159	231	198	146	150	2	234	193	153	252
15	175	130	115	122	201	74	50	173	254	223	121	95	1	38	217	166
16	24	149	76	26	116	240	67	85	10	180	196	144	11	59	108	106

Table 10 S-Box-4 using proposed chaotic PRNG with *Confg* equal to 11

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	243	206	222	218	10	117	13	240	110	229	251	200	216	166	132	120
2	85	101	18	194	68	209	143	50	138	188	32	221	73	53	106	82
3	123	30	213	89	214	184	15	69	104	25	159	56	8	40	178	145
4	142	205	37	226	108	136	203	233	34	163	135	174	212	20	118	137
5	27	168	156	207	246	1	141	211	95	189	71	91	193	154	116	177
6	190	124	97	128	172	61	3	19	234	139	35	245	247	153	114	63
7	228	78	122	75	70	76	38	94	33	115	62	45	152	16	80	66
8	165	160	7	161	90	83	175	67	130	148	86	219	220	167	225	144
9	28	198	249	239	158	237	98	88	49	87	113	65	147	2	252	131
10	9	253	197	238	12	201	11	140	192	185	111	248	173	39	187	41
11	241	105	224	22	250	126	103	217	74	164	44	29	36	0	150	60
12	54	223	119	210	244	121	176	64	215	169	208	59	133	17	43	46
13	93	57	236	171	195	199	191	196	14	72	180	24	52	146	254	235
14	42	232	21	227	47	99	96	181	26	186	77	129	179	92	157	109
15	125	48	230	242	55	84	204	5	102	134	81	162	183	255	127	202
16	31	149	100	79	4	58	182	23	112	6	51	151	155	231	170	107

4 are 0.4063, 0.6094, and 0.5005 respectively. Its average value corresponding to S-boxes is very closer to 0.5. Thus, the property of SAC for proposed S-Boxes is satisfied.

To evaluates the security strength of S-Box, Bits Independence Criterion (BIC) is also important. For the S-boxes, the static pattern among output vectors and no dependency on each other is ensured by the BIC parameters. The corresponding BIC non-linearity for the S-box-1, -2, -3, and -4 has been tabulated in Table [19, 20, 21, 22]. Further, the BIC non-linearity value of 102.5714, 103.1429, 102.8571, and 103.2143 also has been calculated for the S-box-1, -2, -3, and -4 respectively. The SAC properties are also measured by the dynamic distance (DD) (Ishfaq 2018) and it is satisfied only when there is a small integral value for dynamic distance. The DD for S-Box-1, -2, -3, and -4 have been tabulated in Table [11, 12, 14, 15]. The calculated average values of DD for S-box-1, -2, -3 and -4 are 5.3125, 5.125, 4.34375 and 4.625 respectively which holds a better inclination for the fulfill the BIC criterion.

Table 10 illustrates the comparison of proposed S-boxes in terms of the property of Bijection, Nonlinearity, SAC, and BIC Non-Linearity with the existing literature. This table helps to conclude the important criterion such as Bijective, Balanced, Non-linearity, and Avalanche Criteria. It has been satisfied by these boxes. Further, the average value of non-linearity of S-box-1, -2, -3, and -4 are 103.75, 104.25, 104.00, and 105.00 correspondingly, which indicates the value of proposed S-boxes is much better than that reported in the literature (Cassal-Quiroga & Campos-Cantón, 2020). It has been observed that the expected bijection value of 128 has been fulfilled by the S-Boxes. Moreover, S-Box-1, -2, -3, and -4 have mean SAC value of 0.500016, 0.504894, 0.503669 and 0.5005 respectively that is much closer to 0.5. The BIC-nonlinearity average values are 102.5714, 103.1429, 102.8571, and 103.2143 for S-box-1, -2, -3, and -4 which reveal the betterment of S-boxes.

Table 11 Dynamic Distance (DD) of S-box-1

2	12	2	2	6	8	4	2
6	8	2	6	12	2	6	10
6	6	4	6	0	10	6	2
6	4	10	0	6	4	12	0
8	10	8	6	14	2	10	2
4	10	2	2	2	12	4	4
2	2	2	10	4	2	2	0
4	8	0	10	4	8	4	6

Table 12 Dynamic Distance Table of S-box-2

4	4	4	0	0	2	2	6
2	2	2	6	2	6	10	6
0	6	0	8	2	4	18	8
2	6	4	8	12	0	6	6
4	2	2	14	10	10	8	2
4	4	10	4	14	2	0	0
12	2	8	6	6	8	4	2
6	2	6	6	6	10	2	4

Table 13 Dynamic Distance Table of S-box-2

4	4	4	0	0	2	2	6
2	2	2	6	2	6	10	6
0	6	0	8	2	4	18	8
2	6	4	8	12	0	6	6
4	2	2	14	10	10	8	2
4	4	10	4	14	2	0	0
12	2	8	6	6	8	4	2
6	2	6	6	6	10	2	4

Table 14 Dynamic Distance Table of S-box-3

2	4	2	2	2	10	0	12
2	6	4	8	2	8	6	8
4	2	6	4	2	6	2	6
12	2	0	2	6	0	2	0
14	4	10	4	0	2	6	10
4	4	4	0	6	4	2	10
0	0	0	2	12	4	2	2
2	0	8	6	4	2	10	6

Table 15 Dynamic Distance Table of S-box-4

0	2	8	2	10	2	4	4
6	12	2	2	4	8	6	16
6	0	4	0	2	8	14	4
2	6	2	10	0	6	4	2
8	10	0	4	6	8	2	8
2	8	10	2	4	2	0	0
10	8	4	2	0	8	4	4
10	2	2	2	2	2	4	0

Table 16 SAC criterion result of the generated S-box-1

0.4844	0.5938	0.4844	0.4844	0.5469	0.5625	0.5313	0.4844
0.5469	0.4375	0.5156	0.4531	0.4063	0.5156	0.5469	0.4219
0.5469	0.5469	0.5313	0.5469	0.5	0.5781	0.5469	0.4844
0.5469	0.4688	0.4219	0.5	0.5469	0.5313	0.4063	0.5
0.5625	0.4219	0.5625	0.5469	0.3906	0.5156	0.5781	0.5156
0.4688	0.5781	0.4844	0.4844	0.5156	0.4063	0.4688	0.5313
0.5156	0.4844	0.5156	0.4219	0.4688	0.5156	0.4844	0.5
0.4688	0.5625	0.5	0.4219	0.4688	0.4375	0.5313	0.4531

Table 17 SAC criterion result of the generated S-box-3

0.5156	0.5313	0.4844	0.5156	0.5156	0.5781	0.5	0.4063
0.5156	0.5469	0.5313	0.5625	0.4844	0.5625	0.4531	0.4375
0.4688	0.5156	0.5469	0.4688	0.4844	0.4531	0.5156	0.5469
0.4063	0.5156	0.5	0.5156	0.4531	0.5	0.4844	0.5
0.3906	0.4688	0.5781	0.5313	0.5	0.5156	0.5469	0.5781
0.4688	0.5313	0.5313	0.5	0.4531	0.5313	0.5156	0.4219
0.5	0.5	0.5	0.5156	0.5938	0.5313	0.4844	0.5156
0.5156	0.5	0.5625	0.4531	0.4688	0.4844	0.5781	0.4531

Table 18 SAC criterion result of the generated S-box-4

0.5	0.5156	0.5625	0.4844	0.4219	0.5156	0.4688	0.4688
0.4531	0.4063	0.5156	0.4844	0.4688	0.5625	0.5469	0.625
0.5469	0.5	0.5313	0.5	0.4844	0.4375	0.6094	0.5313
0.5156	0.5469	0.4844	0.5781	0.5	0.5469	0.4688	0.5156
0.5625	0.4219	0.5	0.5313	0.4531	0.5625	0.4844	0.4375
0.4844	0.4375	0.5781	0.5156	0.5313	0.4844	0.5	0.5
0.4219	0.4375	0.5313	0.4844	0.5	0.4375	0.4688	0.5313
0.4219	0.5156	0.5156	0.5156	0.4844	0.5156	0.4688	0.5

Table 19 BIC Non-linearity criterion of S-box-1

0	98	100	104	102	106	108	106
98	0	100	102	104	98	100	104
100	100	0	102	104	96	100	98
104	102	102	0	106	102	106	100
102	104	104	106	0	104	104	108
106	98	96	102	104	0	102	106
108	100	100	106	104	102	0	102
106	104	98	100	108	106	102	0

Table 20 BIC Non-linearity criterion of S-box-2

0	104	104	104	102	100	102	106
104	0	104	104	98	106	102	104
104	104	0	102	106	104	104	106
104	104	102	0	100	102	108	104
102	98	106	100	0	102	98	104
100	106	104	102	102	0	100	102
102	102	104	108	98	100	0	106
106	104	106	104	104	102	106	0

Table 21 BIC Non-linearity criterion of S-box-3

0	106	100	102	106	104	102	102
106	0	100	102	106	106	100	104
100	100	0	106	100	104	96	106
102	102	106	0	98	102	104	104
106	106	100	98	0	106	104	102
104	106	104	102	106	0	98	106
102	100	96	104	104	98	0	104
102	104	106	104	102	106	104	0

Table 22 BIC Non-linearity criterion of S-box-4

0	106	100	106	104	100	102	104
106	0	106	104	104	104	100	102
100	106	0	104	106	104	108	98
106	104	104	0	100	104	96	104
104	104	106	100	0	106	102	102
100	104	104	104	106	0	108	102
102	100	108	96	102	108	0	104
104	102	98	104	102	102	104	0

Table 23 Comparison of our S-boxes and other S-boxes used in typical block ciphers.

		Bijection	Nonlinearity			SAC			BIC Non-Linearity
			Min.	Max.	Average	Min.	Max.	Average	
(Cassal-Quiroga & Campos-Cantón, 2020)	S-box-1	128	96	104	101.75	0.3906	0.5781	0.5012	103.42
	S-box-2	128	96	108	102.25	0.4219	0.6094	0.5059	103.50
(Gupta & Chauhan, 2021)	S-box-1	128	98	108	103.7500	0.4063	0.5938	0.507583	103.7857
	S-box-2	128	94	108	100.5000	0.3906	0.6094	0.498792	102.9286
Proposed	S-box-1	128	100	108	103.75	0.3906	0.5938	0.500016	102.5714
	S-box-2	128	102	106	104.25	0.3906	0.6406	0.504894	103.1429
	S-box-3	128	100	106	104.00	0.3906	0.5781	0.503669	102.8571
	S-box-4	128	102	110	105.00	0.4063	0.6094	0.5005	103.2143

CONCLUSION

This paper summarizes the design and evaluation of the new four S-boxes for subbyte operation in image encryption applications and estimates their strength using the following parameters: Dynamic Distance, BIC non-linearity, Bijective, Non-linearity, Strict Avalanche Criterion (SAC), and Balanced criterion. The S-box matrix is designed by a new reconfigurable 3D-Chaotic PRNG. This PRNG is designed using four different 3D chaotic systems i.e. Lorenz, Chen, Lu, and Pehlivan's chaotic systems. This reconfigurable architecture of PRNG exploits the ODEs of these four attractors that fit all four chaotic systems in a single circuit. The novelty of this PRNG is multiplierless VLSI architecture. That offers relatively better performance. To demonstrate the hardware performance, the PRNG circuit is implemented in Virtex-5 (XC5VLX50T) FPGA and finds the timing performance which generates the 24-bit random number at 96.438-MHz. The area of FPGA is occupied by only 16.66%, 1.08%, 0.33%, and 1.15% of the available DSP blocks, slice LUTs, slice registers and slices respectively. Finally, the proposed four different S-box matrixes fulfill the following criteria: Dynamic Distance, BIC non-linearity, Bijective, Non-linearity, Strict Avalanche Criterion (SAC), and Balanced criterion. Therefore, it can conclude that the proposed S-boxes are used for secure image encryption algorithms.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Ahmad, M. and E. A. Alsolami, 2020 Evolving dynamic s-boxes using fractional-order hopfield neural network based scheme. Entropy **22**.

Akgul, A., C. Arslan, and B. Arıcıoğlu, 2019 Design of an interface for random number generators based on integer and fractional order chaotic systems. volume 1, pp. 1–18.

Alawida, M., A. Samsudin, and J. S. Teh, 2020 Enhanced digital chaotic maps based on bit reversal with applications in random bit generators. Inf. Sci. **512**: 1155–1169.

Alçın, M., İ. Pehlivan, and İ. Koyuncu, 2016 Hardware design and implementation of a novel ann-based chaotic generator in fpga. Optik **127**: 5500–5505.

Alhadawi, H. S., D. Lambić, M. F. B. Zolkipli, and M. Ahmad, 2020 Globalized firefly algorithm and chaos for designing substitution box. J. Inf. Secur. Appl. **55**: 102671.

Artuğer, F., 2023 A new s-box generator algorithm based on 3d chaotic maps and whale optimization algorithm. Wireless Personal Communications **131**: 1–19.

Artuğer, F. and F. Özkaynak, 2022a A method for generation of substitution box based on random selection. Egyptian Informatics Journal **23**: 127–135.

Artuğer, F. and F. Özkaynak, 2022b Sbox-cga: substitution box generator based on chaos and genetic algorithm. Neural Computing and Applications **34**: 1–9.

Cassal-Quiroga, B. B. and E. Campos-Cantón, 2020 Generation of dynamical s-boxes for block ciphers via extended logistic map. Mathematical Problems in Engineering **2020**: 1–12.

ElSafty, A. H., M. F. Tolba, L. A. Said, A. H. Madian, and A. G. Radwan, 2021 Analog integrated circuits and signal processing. Hardware realization of a secure and enhanced s-box based speech encryption engine **106**: 385–397.

G. Di Patrizio Stanchieri, E. P., A. De Marcellis and M. Faccio, 2019 A true random number generator architecture based on a reduced number of fpga primitives. AEU - Inte. J. Electron. Commun. **105**.

Garcia-Bosque, M., A. Pérez-Resa, C. Sánchez-Azqueta, C. Aldea, and S. Celma, 2019 Chaos-based bitwise dynamical pseudorandom number generator on fpga. IEEE Transactions on Instrumentation and Measurement **68**: 291–293.

Garcia-Bosque, M., A. Pérez-Resa, C. Sánchez-Azqueta, C. Aldea, and S. Celma, 2018 A new technique for improving the security of chaos based cryptosystems. In *2018 IEEE International*

Symposium on Circuits and Systems (ISCAS), pp. 1–5.
Garipcan, A. M. and E. Erdem, 2020 A trng using chaotic entropy pool as a post-processing technique: analysis, design and fpga implementation. Analog Integr. Circuits Signal Process. **103**: 391–410.
Gupta, M. and R. Chauhan, 2020 Efficient hardware implementation of pseudo-random bit generator using dual-clcg method. Journal of Circuits, Systems and Computers **30**.
Gupta, M. D. and R. K. Chauhan, 2021 Secure image encryption scheme using 4d-hyperchaotic systems based reconfigurable pseudo-random number generator and s-box. Integr. **81**: 137–159.
Gupta, M. D. and R. K. Chauhan, 2022 "hardware efficient pseudo-random number generator using chen chaotic system on fpga. J. Circuits, Syst. Comput. **31**: 2250043.
H. S. Alhadawi, S. M. I., M. F. Zolkipli and D. Lambi, 2019 Designing a pseudorandom bit generator based on lfsrs and a discrete chaotic map. Cryptologia **43**: 190–210.
Ishfaq, F., 2018 *A MATLAB Tool for the Analysis of Cryptographic Properties of S-boxes*. MATLAB Tool for the Analysis of Cryptographic Properties of S-boxes.
Khan, H., M. M. Hazzazi, S. S. Jamal, I. Hussain, and M. Khan, 2022 New color image encryption technique based on three-dimensional logistic map and grey wolf optimization based generated substitution boxes. Multimedia Tools and Applications **82**: 1–22.
Khan, M., T. Shah, and S. I. Batool, 2016 Construction of s-box based on chaotic boolean functions and its application in image encryption. Neural Computing and Applications **27**: 677–685.
Lambić, D. and M. Nikolic, 2019 New pseudo-random number generator based on improved discrete-space chaotic map. Filomat **33**: pp. 2257–2268.
Li, Q. and X. S. Yang, 2010 simple method for finding topological horseshoes. A simple method for finding topological horseshoes **20**: 467–478.
Li, S., X. Mou, and C. Yuanlong, 2001 Pseudo-random bit generator based on couple chaotic systems and its applications in stream-cipher cryptography. In *International Conference on Cryptology in India*.
Lorenz, E. N., 1963 Deterministic nonperiodic flow. Journal of the Atmospheric Sciences **20**: 130–141.
Lu, J. and G. Chen, 2002 A new chaotic attractor coined. Int. J. Bifurc. Chaos **12**: 659–661.
L'Ecuyer, P., 2012 Random number generation. in Handbook of Computational Statistics .
Pande, A. and J. Zambreno, 2013 A chaotic encryption scheme for real-time embedded systems: design and implementation. Telecommunication Systems **52**: 551–561.
Pehlivan, I. and Y. Uyaroğlu, 2010 A new chaotic attractor from general lorenz system family and its electronic experimental implementation. Turkish Journal of Electrical Engineering and Computer Sciences **18**: 171–184.
Pehlivan, I. and Y. Uyaroğlu, 2012 A new 3d chaotic system with golden proportion equilibria: Analysis and electronic circuit realization. Comput. Electr. Eng **38**: 285–317.
Rezk, A. A., A. H. Madian, A. G. Radwan, and A. M. Soliman, 2019 Reconfigurable chaotic pseudo random number generator based on fpga. AEU - International Journal of Electronics and Communications .
Rezk, A. A., A. H. Madian, A. G. Radwan, and A. M. Soliman, 2020 Multiplierless chaotic pseudo random number generators. Aeu-international Journal of Electronics and Communications **113**: 152947.
Rukhin, A. L., J. Soto, J. Nechvatal, M. E. Smid, and E. B. Barker, 2000 A statistical test suite for random and pseudorandom number generators for cryptographic applications. volume 2, pp. 1–8.
T. Zhou, G. C. and S. Čelikovský, 2005 Lnikov chaos in the generalized lorenz canonical form of dynamical systems,. Nonlinear Dyn. **39**: 319–334.
Tang, G., X. Liao, and Y. Chen, 2005 A novel method for designing s-boxes based on chaotic maps. Chaos Solitons & Fractals **23**: 413–419.
Wang, X., Ü. Çavusoglu, S. Kaçar, A. Akgul, V.-T. Pham, *et al.*, 2019 S-box based image encryption application using a chaotic system without equilibrium. Applied Sciences **9**: 4.
Wang, Y., Z. Liu, J. Ma, and a. H. He, 2016 pseudorandom number generator based on piecewise logistic map. Nonlinear Dyn. **83**: 2373–2391.
Wang, Y., K. wo Wong, X. Liao, and T. Xiang, 2009 A block cipher with dynamic s-boxes based on tent map. Communications in Nonlinear Science and Numerical Simulation **14**: 3089–3099.
Wolf, A., J. B. Swift, H. L. Swinney, and J. A. Vastano, 1985 A new 3d chaotic system with golden proportion equilibria: Analysis and electronic circuit realization. Phys. D Nonlinear Phenom. **16**: 285–317.
X. Y. Wang, R. L., L. Yang and A. Kadir, 2010 A chaotic image encryption algorithm based on perceptron model. Nonlinear Dyn. **62**: 615–621.
Xu, W., J. Feng, and H. Rong, 2009 Melnikov's method for a general nonlinear vibro-impact oscillator. Nonlinear Anal. Theory, Methods Appl. **71**: 418–426.
Zahid, A. H., A. M. Iliyasu, M. Ahmad, M. M. U. Shaban, M. J. Arshad, *et al.*, 2021 A novel construction of dynamic s-box with high nonlinearity using heuristic evolution. IEEE Access **9**: 67797–67812.
Zamli, K. Z., F. Din, H. S. Alhadawi, S. Khalid, H. Alsolai, *et al.*, 2023 Exploiting an elitist barnacles mating optimizer implementation for substitution box optimization. ICT Express **9**: 619–627.
Zhao, Y., C. Gao, J. Liu, and S. Dong, 2019 A self-perturbed pseudo-random sequence generator based on hyperchaos. volume 4, p. 100023.
Zidan, M. A., A. G. Radwan, and K. N. Salama, 2011 The effect of numerical techniques on differential equation based chaotic generators. ICM 2011 Proceeding pp. 1–4.
Çavusoglu, Ü., S. Kaçar, I. Pehlivan, and A. Zengin, 2017 Secure image encryption algorithm design using a novel chaos based s-box. Chaos Solitons & Fractals **95**: 92–101.
Ü. Çavuşoğlu, A. A. S. J., S. Panahi and S. Kaçar, 2019 A new chaotic system with hidden attractor and its engineering applications: analog circuit realization and image encryption,.

How to cite this article: Gupta, M. D., Chauhan, R. K., and Upadhyay, V. K. Analyses of Reconfigurable Chaotic Systems and their Cryptographic S-box Design Applications. *Chaos Theory and Applications*, 5(3), 219-232, 2023.